Application Report
High Resolution Microstepping Driver

TEXAS INSTRUMENTS

ABSTRACT

This application note details the implementation of a motor driver with an output current regulation function and a MSP430F1612 microcontroller in a combination that allows the motor driver to drive a bipolar stepper motor in microstepping configuration with increased degrees of microstepping.

Table of Contents

List of Figures

List of Tables

Trademarks
All trademarks are the property of their respective owners.

1 Introduction

Bipolar stepper motors are often designed to be driven at full steps. However, it is a known fact that when each step is divided into multiple and smaller steps (referred to as micro steps), better motion quality can be observed. The application of microstepping to practically any stepper application allows for other improvements such as:

1. Better torque response
2. Less vibration
3. Less resonance incidence

High resolution microstepping can be achieved by modulating the VREF analog inputs provided for current regulation setting in the driver. All which needs to be done is to apply an analog waveform to said reference voltage inputs, and the winding current magnitude will be regulated to follow the respective input. Generating this waveform can be accomplished by the utilization of a DAC output such as the ones found on an MSP430 microcontroller. Putting all of this together gives us a series of advantages such as:

1. Degrees of microstepping can be made as large as there is resolution on the DAC block. With a 12 bit DAC, you could have up to 4096 levels of microstepping. Far much more than most applications should require.
2. Any waveform, however intricate, can be encoded into the VREF analog inputs. This could be used to resolve any magnetic non linearity within the stepper motor itself.

In this application note example we will detail the mechanisms put in place to achieve high resolution microstepping by utilizing a microcontroller, such as the MSP430F1612, in combination with a dual H-bridge DC motor drive device with an output current regulation function. The same concept can also be applied to two single H-bridge devices.

2 Full Step vs. Microstep

2.1 The Full Step

Bipolar stepper motors are often designed to be driven at full steps. However, it is a known fact that when each step is divided into multiple and smaller steps (referred to as micro steps), better motion quality can be observed. The application of microstepping to practically any stepper application allows for other improvements such as:

1. Better torque response
2. Less vibration
3. Less resonance incidence

High resolution microstepping can be achieved by modulating the VREF analog inputs provided for current regulation setting in the driver. All which needs to be done is to apply an analog waveform to said reference voltage inputs, and the winding current magnitude will be regulated to follow the respective input. Generating this waveform can be accomplished by the using a DAC output such as the ones found on an MSP430 microcontroller. Putting all of this together gives us a series of advantages such as:

1. Degrees of microstepping can be made as large as there is resolution on the DAC block. With a 12 bit DAC, you could have up to 4096 levels of microstepping. Far much more than most applications should require.
2. Any waveform, however intricate, can be encoded into the VREF analog inputs. This could be used to resolve any magnetic non linearity within the stepper motor itself.

In this application note example we detail the mechanisms put in place to achieve high resolution microstepping by using a microcontroller, such as the MSP430F1612, in combination with a dual H-bridge DC motor driver with an output current regulation function. The same concept can be applied to two single H-bridge devices.

Bipolar Motor and Drive   Polarization Sequencing
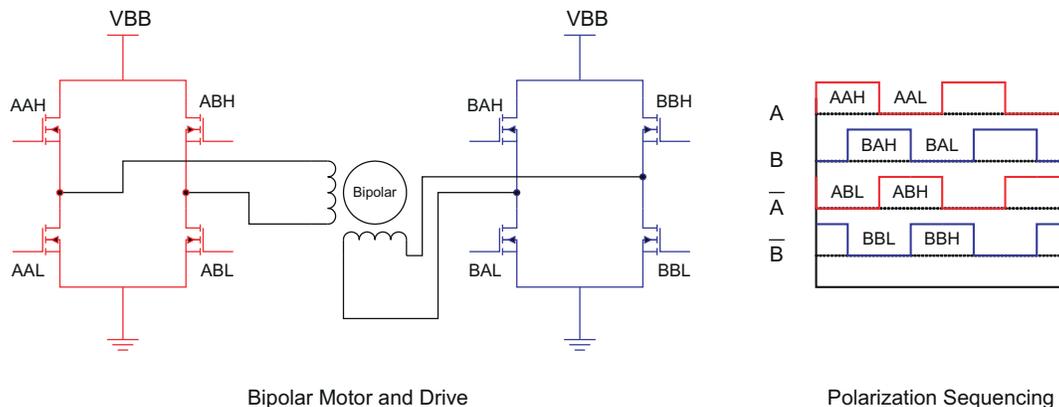
Figure 2-1. Full Step

Stepper motors are brushless motors. In other words, they lack a brush making mechanical contact with a commutator, as housed on the typical brushed DC motor. Because of this, stepper motors must be electronically communicated. The most common way of communicating the rotor into motion is by modulating the magnetic field of each stator electromagnet.

There are two stator electromagnets on any conventional stepper motor, which is why two H Bridges are needed. By modulating both PHASE A and PHASE B pins, to obtain a polarization waveform such as the one depicted in Figure 2-1, the motor will move in full steps.

2.2 The Micro Step

If full steps are all that the application requires, then Figure 2-1 is the optimal solution. However, the great majority of applications can certainly benefit from dividing full steps into smaller steps.

Figure 2-2 shows two full steps on the top and a way to divide each full step into 8 smaller steps. We have chosen a sine wave shape, although in reality this waveform can have any shape the users chooses. The idea behind microsteps is to make the current flowing through the winding different than FULL ON on one direction and FULL ON on the opposite direction. This allows for the rotor to be pushed/pulled in increments rather than full force.
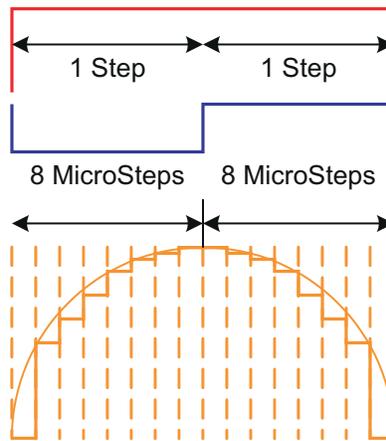
Figure 2-2. Micro Step

Because the DC motor driver has the capability to regulate winding current, and said current is directly proportional to the VREF input, it is possible to embed the prospective waveform at the current side. Figure 2-3 shows the result of embedding said waveform into the winding current, which is the concept behind this application note.
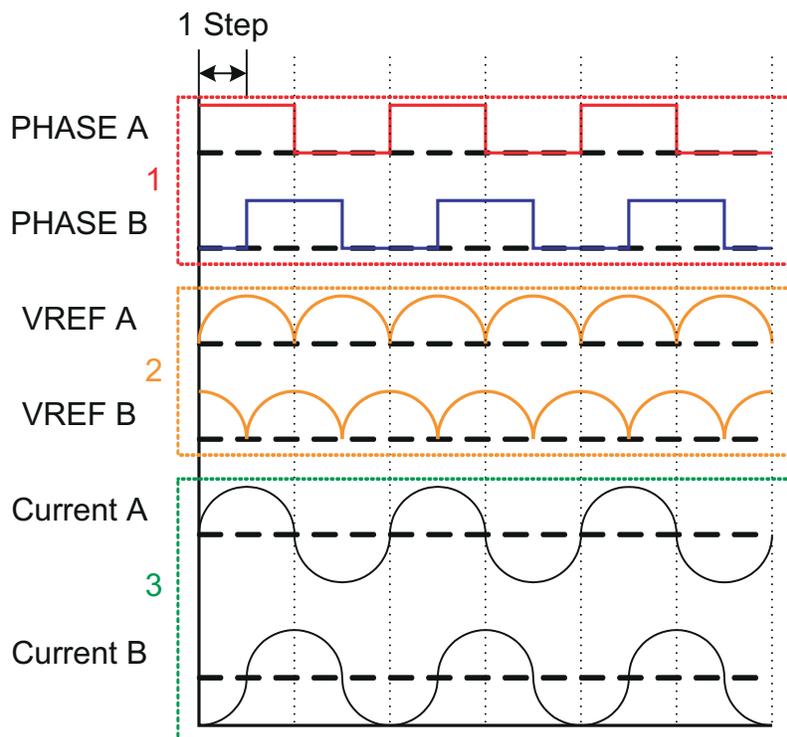

Figure 2-3. Embedded Waveform Into Winding Current

A microcontroller, such as the MSP430F1612 and utilizing GPIO outputs, will drive the motor driver's PHASE inputs in the alternating fashion we saw on previous figures. However, and at the same time, the VREF pins will be modulated by using the two DAC outputs to superimpose a sine wave crest.

As a result, the motor windings see a full sine wave of current being applied to them. Because the current is segmented in smaller steps, the rotor also moves in smaller steps.

3 Application Block Diagram

In essence, the combination portrayed in Figure 3-1 is similar to how a stepper with internal indexer operates. A logic block samples a series of inputs and generates control signals to the dual H Bridge device. Typical internal indexer driver devices represent an encased application and can not be modified. However, by splitting the application into a microcontroller and a driver, more flexibility is obtained.

Figure 3-1 shows how the MSP430F1612 drives the control signals such as PHASE, ENABLE and VREF. ENABLE and PHASE signals are derived from conventional GPIO configured as outputs, whereas the VREF analog signals are derived from both of MSP430F1612's DAC outputs. An interrupt subroutine computes the next step and generates the next microstep.
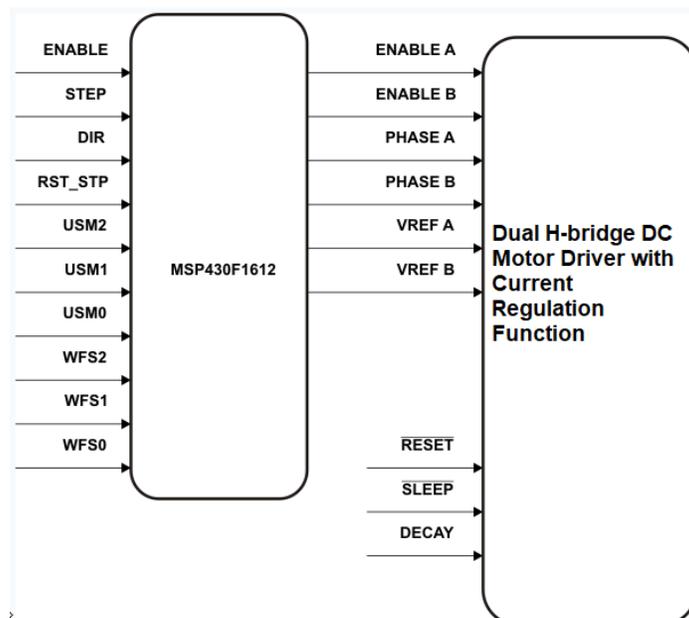
Figure 3-1. Application Block Diagram

3.1 Microstepping Indexer

At the heart of our microstep generation engine is a look-up table that houses up to eight waveforms, In reality, the MSP430F1612 has enough space to hold much more than eight tables, but for this application note and demonstration purposes, eight tables seemed enough.

Each one of these look-up tables is 512 steps deep. Which waveform table is selected is a factor of the three WFS bits.

To generate a step, two values are read from the table and loaded into the DAC registers. For PHASE A to be 90 degrees out of phase with regards to PHASE B, each table item read is located on addresses 256 steps apart from each other (when on 256 microsteps mode). Whether PHASE A lags or leads B depends on the DIR bit. For Clockwise, we chose PHASE A leads B, while for counter clockwise, PHASE A lags PHASE B was chosen. This is totally arbitrary and either way is perfectly acceptable as long as the motor is wired accordingly.
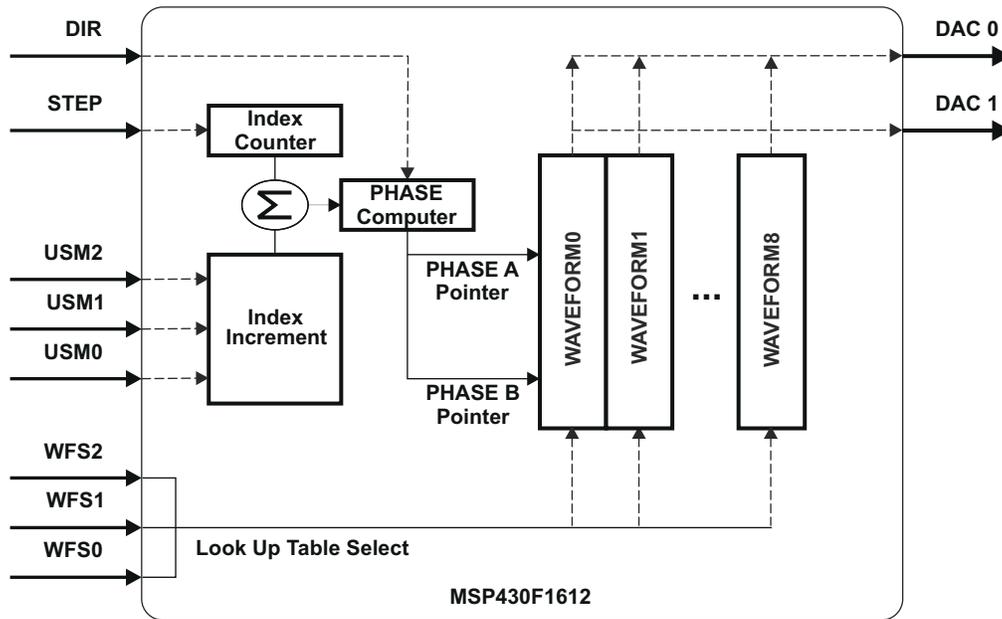
Figure 3-2. Microstepping Indexer

Every time a STEP is issued, a counter is incremented. This counter becomes a pointer to the look-up table. The counter is incremented by 1 if on the highest resolution setting. In this case, maximum degrees of microstepping is obtained. However, to obtain less degrees of microstepping, there is no need to have a smaller look up table as the same large look up table suffices.

To obtain lesser degrees of microstepping, the counter is incremented by a larger factor, that must be base 2. Per example, to obtain 256 degrees of microstepping, the incrementing value is 1 and to obtain 128 degrees of microstepping, the incrementing factor is 2. Subsequent multiplications of 2 for the incrementing factor, take the microstepping degrees down to half of what they were before.

The Index Increment is a factor of the USMx bits. With three bits we have up to 8 different combinations. In this case, the indexer can operate with half step, quad step as well as 8, 16, 32, 64, 128, 256 degrees of microstepping, as depicted in Table 3-1.

Table 3-1. Microstepping Operation

Degrees of Microstepping	Divider Factor (Lookup Table Index Increment)
256	1
128	2
64	4
32	8
16	16
8	32
4	64
2	128

3.2 Microcontroller Side Control Signals

ENABLE: Microcontroller timer input configured to capture and interrupt on both edges. Signals the microcontroller whether the H Bridges should be powered up or not. Microcontroller remains powered on low power mode when ENABLE is LO and will operate accordingly when ENABLE is HI.

STEP: Microcontroller timer input configured to capture and interrupt on rising edges. A transition from LO to HI tells the microcontroller to issue a new step. Transitions from HI to LO are ignored, although it is the user's prerogative to code this so that both transitions generate a step.

DIR: Microcontroller timer input configured to capture and interrupt on both edges. Selects the bipolar motor direction of rotation. LO implies counter clockwise rotation, while HI selects clockwise rotation. This was arbitrarily chosen and can be changed according to preference.

STP_RST: Microcontroller timer input configured to capture and interrupt on both edges. Clears the internal indexer and disables the H Bridge. When STP_RST is made LO, the indexer look up table is returned to pointer 0 and the H Bridge is disabled. When STP_RST returns to HI, the H Bridges are enabled and motion starts from look-up table position 0.

USMx: Microcontroller timer input configured to capture and interrupt on both edges. Selects the 8 possible degrees of microstepping resolution. In this application note, half step, quad step, as well as 8, 16, 32, 64, 128 and 256 degrees of microstepping were coded.

WFSx: Microcontroller interrupt input (GPIO). Waveform Select bits. Select from the eight internal look up tables.

3.3 Microstepping Engine Algorithm
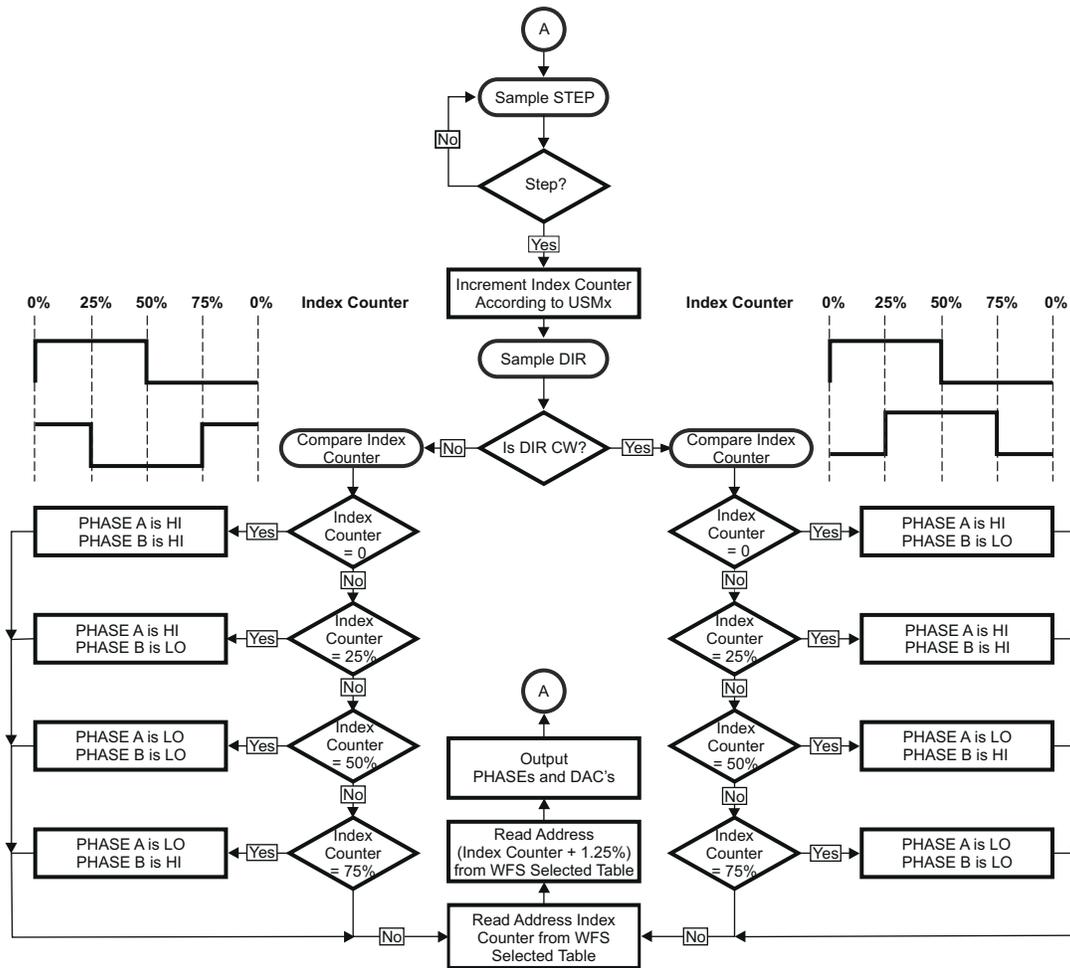
Figure 3-3. Microstepping Engine Algorithm

4 Example Plots

On the following images, the Channel Legend applies:

- CH1: VREF A
- CH2: PHASE A Pin
- CH3: Winding A Current
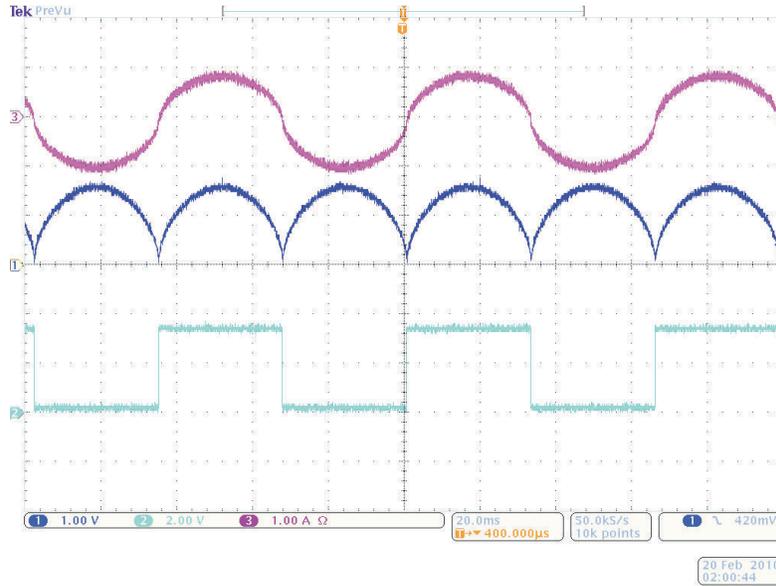
Figure 4-1. Squared Sine Wave with 256 Degrees of Microstepping
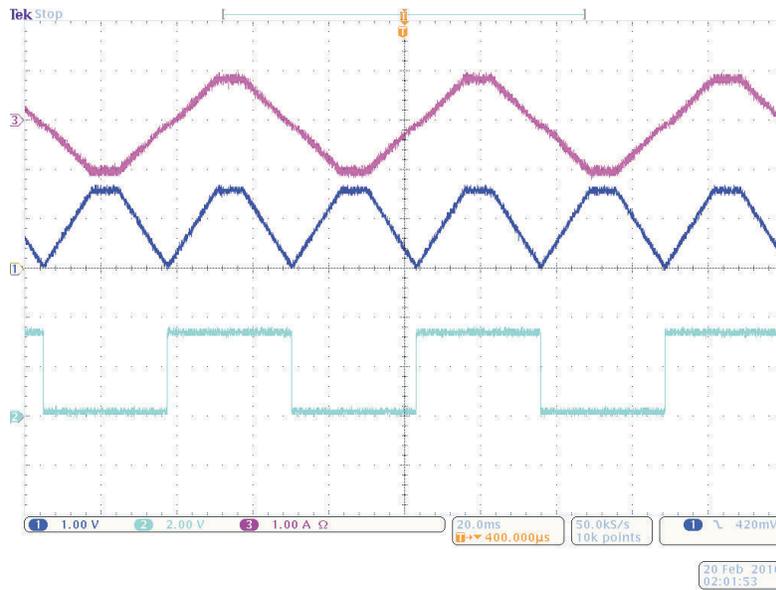
Figure 4-2. Trapezoid Waveshape with 256 Degrees of Microstepping
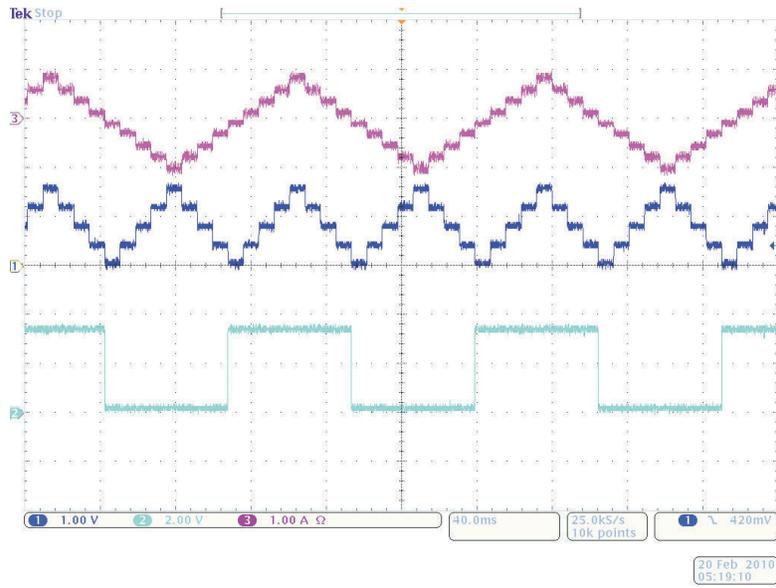
Figure 4-3. Triangular Waveshape with 4 Degrees of Microstepping

5 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

IMPORTANT NOTICE AND DISCLAIMER